On update schedules and dynamics of Boolean networks

Mathilde Noual

M1 internship account

2007-2008

On update schedules and dynamics of boolean networks

Mathilde Noual

Internship done at the university of Concepción under the direction of Julio Aracena

Contents

1 Introduction

One of the common applications there is of boolean networks is the modelisation of genetic networks. Once the architecture of a genetic network is known, it may be interesting to study its dynamics in order to understand better how the network works. Identifying the attractors of such a system (that is, its fixed points and limit cycles) may indeed help us to explain phenomenons such as cellular divisions, destructions and differenciation. In spite of its rather high level of abstraction, the modelisation of genetic networks by means of boolean networks may turn out to be very useful because it makes it possible to study the limit behaviors of such biological systems.

The dynamical properties of a boolean network rely on several different parameters. During this internship, we have especially been looking at three of them which are very closely tied to the definition itself of a boolean network : the structure of a network, the type of local activation functions, and the update schedules. In other words, the dependencies between the elements of a network, the way each one of these elements is activated as a function of the other elements of the network and the order according to which the states of the elements of the network are updated.

The central question that has directed our work naturally followed from the results presented in [1]. It is related to the idea of the robustness of a boolean network with respect to perturbations of its update schedule. On one hand, we have studied these update schedules in terms of *signed digraphs* (formalism introduced in [1]) and on the other hand we have examined particular boolean networks whose dynamical behaviors were identical without the update schedule of the networks being a direct cause of it. This paper starts with a series of definitions that we will be using extesively later on. Section 3 introduces the problems we have worked on during this internship and the last two sections present the results that we have obtained and the observations that we have made.

2 Definitions

A genetic network is represented by a digraph $G = (V, A)$. An arc (i, j) means that the state of the gene or vertice j depends on the state of the vertice i. The state of a vertice i is represented by a boolean variable $x_i \in \{0, 1\}$.

Notations 2.1

- *If G is a digraph, we will refer to the set of its vertices as $V(G)$ and to the set of its arcs as $A(G)$ when these two sets have not already been named.*
- *Also, in the sequel, we will write $[\![a, b]\!] = \{a, \ldots, b\}$ and $[\![a, b[\![= \{a, \ldots, b-1\}$, for any integers a and b.*
- *$G = (V, A)$ being a digraph and $i \in V$ one of its vertices, $N(i)$ denotes the set $\{j \in V \mid (j, i) \in A\}$. If $j \in N(i)$, we say that i depends on j.*

Definition 2.2 *An* **update schedule** *of the vertices of a digraph $G = (V, A)$ such that $|V| = n$ is a function $s : V \to [\![1, n]\!]$ such that the set of all images of the vertices of V is a set of consecutive integers containing the integer 1.*

If $\forall i \in V,\ s(i) = 1$, the update schedule is said to be **parallel**. *In this case, we will write $s = s_p$. If s is a permutation of the set $\{1, \dots, n\}$, s is said to be* **sequential**. *And in all other cases, s is said to be* **block sequential**.

As mentionned in [4], the number of update schedules associated to a given digraph of order n is equal to the number of ordered partitions of a set of size n, that is

$$\mathcal{I}_n = \sum_{k=0}^{n-1} \binom{n}{k} \mathcal{I}_k.$$

Notation 2.3 *Let $G = (V, A)$ be a digraph and s an update schedule. We define $B_t^s = \{i \in V \mid s(i) = t\}$.*

Definition 2.4 *Let $G = (V, A)$ be a digraph and s an update schedule, we define the function $sign_s : A \to \{-, +\}$ in the following way :*

$$\forall (j,i) \in A, sign_s(j,i) = \begin{cases} + & \text{if } s(j) \geq s(i) \\ - & \text{if } s(j) < s(i). \end{cases}$$

An arc $a \in A$ such that $sign_s(a) = +$ is called a **positive arc** *and an arc $a \in A$ such that $sign_s(a) = -$ is called a* **negative arc**. *Also, a set of positive (resp. negative) arcs will be said to be a positive (resp. negative) set of arcs. Labeling every arc a of A by $sign_s(a)$, we obtain a* **signed digraph** *denoted by G^s. If this signed digraph contains only positive (resp. negative) arcs, we will call it a positive (resp. negative) signed digraph.*

Notation 2.5 *We write $N_s^+(i) = \{j \in V \mid sign_s(j,i) = +\}$ and $N_s^-(i) = \{j \in V \mid sign_s(j,i) = -\}$. Thus, we have $N(i) = N_s^+(i) \cap N_s^-(i)$.*

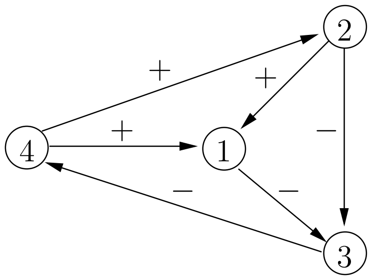


Figure 1: A digraph $G = (V, A)$ signed by the function $sign_s$ where $\forall i \in V = \{1, \dots, 4\}$, $s(i) = i$ (this example is taken from the article [1]).

Definition 2.6 *Let $G = (V, A)$ be a digraph such that $|V| = n$, the* **local activation function** *of a vertice i is a function $f_i : \{0,1\}^n \to \{0,1\}$ such that if $j \in N(i)$, then $\exists x_1, \dots x_{j-1}, x_{j+1}, \dots x_n \in \{0,1\}^{n-1}$,*

$$f_i(x_1, \dots, x_{j-1}, 0, x_{j+1}, \dots, x_n) \neq f_i(x_1, \dots, x_{j-1}, 1, x_{j+1}, \dots, x_n).$$

and if $j \notin N(i)$, then f_i is independant of its j^{th} coordinate. Let s be an update schedule of the vertices of G. We define the function f_i^s verifying $\forall x = x_1 \dots, x_n \in \{0,1\}^n$, $f_i^s(x) = f_i(g_{i,1}^s(x), \dots, g_{i,n}^s(x))$ where

$$g_{i,j}^s(x) = \begin{cases} x_j & si\ s(j) \geq s(i) \text{ i.e. if } sign_s(j,i) = +, \\ f_j^s(x) & si\ s(j) < s(i) \text{ i.e. if } sign_s(j,i) = -. \end{cases}$$

Definition 2.7 *We say a function $f : \{0,1\}^n \to \{0,1\}$ is* **symmetrical** *if $\forall x_1, \dots, x_n \in \{0,1\}^n$, $\forall i < j \in [\![1,n]\!]$, $f(x_1, \dots, x_i, \dots, x_j, \dots, x_n) = f(x_1, \dots, x_j, \dots, x_i, \dots, x_n)$. If f is symmetrical we will sometimes write $f(x) = f(x_j \mid j \in [\![1,n]\!])$.*

In the sequel, all local activation functions will be supposed to be symmetrical.

Definition 2.8 *Let G be digraph such that $|V(G)| = n$, s an update schedule and $\{f_i \mid i \in V\}$ a set of local activation functions. A* **global activation function** *is a function $F : \{0,1\}^n \to \{0,1\}^n$ such that $\forall x \in \{0,1\}^n$, $F(x) = (f_1^s(x), \dots, f_n^s(x))$. Thus, $F^s(x)_i = f_i^s(x) = f_i(g_{i,1}^s(x), \dots, g_{i,n}^s(x))$.*

And if f_i is symmetrical we write

$$F^s(x)_i = f_i(x_j \mid j \in N_s^+(i)\ ;\ f_j^s(x) \mid j \in N_s^-(i)).$$

Definition 2.9 *A* **boolean network** *is a triplet $R = (G, F, s)$ where G is a digraph, F is a global activation function and s an update schedule.*

We will say that two networks $R_1 = (G, F, s_1)$ and $R_2 = (G, F, s_2)$ have the same dynamics if $F^{s_1} = F^{s_2}$.

Definition 2.10 *The set of states $\{0,1\}^n$ being finite, a boolean network can only have two different types of limit dynamical behavior which we call the* **attractors** *of the network:*

- *a* **fix point***, is a point $x \in \{0,1\}^n$ such that $F^s(x) = x$,*
- *a (dynamical)* **cycle** *of length $l > 1$, is a sequence of points $[x^0, \dots, x^{l-1}, x^l]$ such that $x^l = x^0$, $\forall t, r \in [\![1,l]\!]$, $x^t \neq x^r$ and $F^s(x^t) = x^{t+1}$.*

3 Preliminary results and motivations

The following result is taken from the article [1].

Theorem 3.1 *Let $R_1 = (G, F, s_1)$ and $R_2 = (G, F, s_2)$ be two boolean networks that differ only by their update schedule. If $G^{s_1} = G^{s_2}$, then R_1 and R_2 have the same dynamics.*

Remark 3.2 *If* $t^* = min\{s_1(i) \mid \exists j \in N(i),\ sign_{s_1}(j,i) \neq sign_{s_2}(j,i)\}$, *we even have that* $\forall i \in B_t^{s_1}$, $t < t^*$, $f_i^{s_1} = f_i^{s_2}$. *Indeed, let* i *be such a vertice. Since* $\forall j \in N(i)$, $sign_{s_1}(j,i) = sign_{s_2}(j,i)$ *by induction on* $s_1(i)$ ($j \in N_{s_1}^-(i) \Rightarrow s_1(j) < s_1(i)$)*, the following holds*

$$\begin{array}{rcl} f_i^{s_1}(x) & = & f_i(x_j \mid j \in N_{s_1}^+(i)\ ;\ f_j^{s_1}(x) \mid j \in N_{s_1}^-(i)) \\ & = & f_i(x_j \mid j \in N_{s_2}^+(i)\ ;\ f_j^{s_2}(x) \mid j \in N_{s_2}^-(i)) \quad = \quad f_i^{s_2}(x) \end{array}$$

Theorem 3.1 allows us to define *equivalence classes with respect to signed digraphs* : if s is an update schedule of the vertices of a digraph G, we write $[s]^{GS}$ the set of update schedules s' such that $s \overset{GS}{\sim} s'$, *i.e.*, $G^s = G^{s'}$. Thus, an equivalence class, $[s]^{GS}$, is a set of update schedules that all yeild the same signature of a digraph G. Also, we extend this notation to boolean networks : $R_1 = (G, F, s_1)$ and $R_2 = (G, F, s_2)$ being two boolean networks that differ only by their update schedules, we write $R_1 \overset{GS}{\sim} R_2$ if and only if $s_1 \overset{GS}{\sim} s_2$.

During this internship, we have taken interest in two distinct series of questions. Section 4 deals with the first one. More precisely, it deals with the equivalence relation $\overset{GS}{\sim}$. By studying the equivalence classes of $\overset{GS}{\sim}$, we have worked on a characterisation of the robustness of boolean networks with respect to their update schedules. The bigger the sizes of the classes $[\cdot]^{GS}$ and the smaller their number, the more the networks can be considered as robust with respect to their update schedules.

The second series of questions that prompted section 5 of this account, concerns the equivalence relation $\overset{D}{\sim}$ defined below :

$$R_1 \overset{D}{\sim} R_2 \quad \Leftrightarrow \quad \forall x \in \{0,1\}^{|V(G)|},\ \ F^{s_1}(x) = F^{s_2}(x)$$

where $R_1 = (G, F, s_1)$ and $R_2 = (G, F, s_2)$. The equivalence classes for this relation are written[1] $[R]^D$. Obviously, according to theorem 3.1, it holds that $R_1 \overset{GS}{\sim} R_2 \Rightarrow R_1 \overset{D}{\sim} R_2$. However, the example given in figure 3 shows that it is possible to have $G^s \neq G^{s'}$ and $F^s = F^{s'}$, *i.e.*, $R_1 \overset{D}{\sim} R_2$ but not $R_1 \overset{G}{\sim} R_2$ (where R_1 and R_2 differ only by their update schedules). Thus, the converse of theorem 3.1 is not true. The very first question raised during this internship period was about the conditions that would allow us to formulated a kind of converse proposition to this theorem. The observations made following this question are described in section 5.

4 On signed digraphs

In this section, we study the relation $\overset{GS}{\sim}$ and the signatures of a given digraph G. First, we give a characterisation of the signature functions $sign_G : A(G) \to \{+, -\}$ that indeed

[1] When there will be no ambiguity on the digraph G, neither on the function F, given *a priori*, we will also allow ourselves to write $[s]^D$

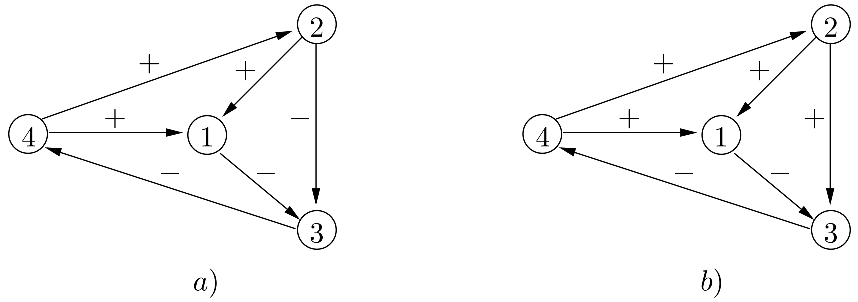


Figure 2: Two boolean networks $R_1 = (G, F, s_1)$ and $R_2 = (G, F, s_2)$ with the same underlying digraph G and the same underlying local activation functions all of which being equal to the boolean function OR. These two networks have different signed digraphs but their dynamics are identical. Let us write $s = x'_1 x'_2 \dots x'_n$ to denote the update schedule of a set of n vertices numbered from 1 to n such that $s(i) = x'_i$, $\forall i \in \{1, \dots, n\}$. *a.* The network R_1. $s_1 = 1234$ *b.* The network R_2. $s_2 = 1324$. Incidently, note that $[s_1]^{GS} = \{1234, 1123\}$, $[s_2]^{GS} = \{1324, 1323, 1223\}$ and $[s_1]^D = [s_1]^{GS} \cup [s_2]^{GS}$.

correspond to the signature functions that are induced by update schedules. Then, we examine update schedules s that, precisely, are such that $sign_G = sign_s$. The section ends with some observations that where made to help determine the number of $[\cdot]^{GS}$ classes and eventually to enumerate them as well.

We first study digraphs G signed by a signature function $sign_G$ *a priori* independant of all update schedule. The non-signed digraph underlying G is denoted by $\mathcal{NS}(G)$.

Definition 4.1 *In the sequel, we say that* G *is a* **possible signed digraph** *if there exists an update schedule* s *such that* $\forall a \in A(G)$, $sign_G(a) = sign_s(a)$. *We call it an* **impossible signed digraph** *otherwise.*

We will also be mentioning possible or impossible **signatures** *designating a function* $sign_G$ *that gives a sign to each arc of a digraph making it a possible or an impossible signed digraph.*

Notation 4.2 *Let* G *be a signed digraph.* $\mathcal{I}(G)$ *denotes the set of update schedules* s *of the set of vertices of* $\mathcal{NS}(G)$ *such that* $(\mathcal{NS}(G))^s = G$. G *is thus possible if and only if* $|\mathcal{I}(G)| > 0$.

4.1 Possibles signed digraphs

The goal of this section is to determine which are the possible signatures of a digraph. First, let us give a few more definitions.

Definition 4.3 *Let* $G = (V, A)$ *be a digraph signed by the function* $sign_G$. *We call* **reducted digraph** *associated to* G*, and write* $\mathcal{RD}(G)$*, to refer to the digraph in which each strongly connected component* $C_i = \{v_{i_1}, \dots, v_{i_l}\} \subseteq V$ *such that*

$$a = (v_{i_j}, v_{i_k}) \in A \cap C_i \Rightarrow sign_G(a) = +$$

is reduced to one vertice. That is, if $\{C_i \mid 1 \le i \le k\}$ *is the set of positive strongly connected components of* G *and if* $\mathcal{C} = \bigcup_{1 \le i \le k} C_i$ *is the reunion of all such strongly connected components, then*

$$\begin{array}{rcl} V(\mathcal{RD}(G)) & = & \{C_i \mid 1 \le i \le k\} \cup (V \setminus \mathcal{C}) \quad \textit{and} \\ A(\mathcal{RD}(G))) & = & \{(u,v) \in A \mid u \notin \mathcal{C} \ \textit{ou}\ v \notin \mathcal{C}\} \ \cup \\ & & \{(v, C_i) \mid 1 \le i \le k,\ v \notin \mathcal{C}\} \ \cup \ \{(C_i, v) \mid 1 \le i \le k,\ v \notin \mathcal{C}\} \end{array}$$

We will say that a signed digraph is reduced if it has no positive strongly connected components.

Definition 4.4 *Let* $G = (V, A)$ *be a digraph signed by the function* $sign_G$*. We call* **reoriented graph** *associated to* G*, and write* $\mathcal{RO}(G) = (V, A(\mathcal{RO}(G)))$*, to refer to the digraph in which all negative arcs are inverted :*

$$A(\mathcal{RO}(G)) = \{a \in A \mid sign_G(a) = +\} \ \cup \ \{(v,u) \mid (u,v) \in A \wedge sign_G(u,v) = -\}.$$

In $\mathcal{RO}(G)$*, an arc* (v, u) *such that* $sign_G(u, v) = -$ *is called a* **$>$-arc** *since any update schedule* s *satisfying* $(\mathcal{NS}(G))^s = G$ *must be such that* $s(v) > s(u)$*. Similarily, an arc* (v, u) *such that* $sign_G(u, v) = +$ *is called a* **$\geq$-arc***.*

Let $G = (V, A)$ be a signed digraph. We can obtain $\mathcal{RD}(G)$ in time $\mathcal{O}(|A|)$ with an algorithm that searches for strongly connected components of a digraph. We also can get $\mathcal{RO}(G)$ in time $\mathcal{O}(|A|)$.

Definition 4.5 *Let* G *be a signed digraph. A* **prohibited circuit** *is an (oriented) circuit of* $\mathcal{RO}(G)$ *containing a* $>$*-arc.*

Theorem 4.6 *Let* G *be a digraph signed by the function* $sign_G$*.* G *is possible if and only if* $\mathcal{RO}(G)$ *does not contain any prohibited circuits.*

Preuve Let us suppose that $\mathcal{RO}(G)$ contains a prohibited circuit $C = (v_{i_1}, \ldots, v_{i_p})$ such that $(v_{i_j}, v_{i_{j+1}})$ is a $>$-arc. Then any update schedule s such that $(\mathcal{NS}(G))^s = G$ must satisfy $s(v_{i_j}) > s(v_{i_{j+1}})$. It must also satisfy $s(v_{i_j}) \le s(v_{i_{j+1}})$ since there exists in $\mathcal{RO}(G)$ a walk from $v_{i_{j+1}}$ to v_{i_j}. Thus, we end up with a contradiction.

For the converse of the theorem, first note that if G is an ordinary signed digraph and $H = \mathcal{RD}(G)$ its associated reduced digraph, then G is possible if and only if H is. Indeed, let s be an update schedule such that $(\mathcal{NS}(H))^s = H$. Then, using again the notations used in definition 4.3, the update schedule s' defined below satisfies $(\mathcal{NS}(G))^s = G$:

$$s'(v) = \begin{cases} s(v) & \text{if } v \notin \mathcal{C} \\ s(C_i) & \text{if } v \in C_i. \end{cases}$$

Vice-versa, if H is an impossible signed digraph, then G is obviously also an impossible signed digraph according to the first paragraph of this proof. The fact that a signature is possible or not is thus independant of the presence or absence of positive strongly connected components. The converse of theorem 4.6 comes from the algorithm 1 given below which finds an update schedule corresponding to a given possible, reduced, signed digraph. $\square$

We can notice that if $G = (V, A)$ is a signed digraph, the prohibited circuits of $\mathcal{RO}(G)$ correspond to what we will refer to as **alternating circuits** of G. That is, they coincide with walks of G, $C = (v_0, v_1, \ldots, v_k)$, where $v_0 = v_k$ and either $(v_i, v_{i+1}) \in A$ in which case $sign_G(v_i, v_{i+1}) = +$, either $(v_{i+1}, v_i) \in A$ in which case $sign_G(v_{i+1}, v_i) = -$ (or vice-versa). Among these alternating circuits, are in particular circuits such that $\forall i \in [\![0, k-1]\!]$, $sign_G(v_i, v_{i+1}) = -$ as well as sub-graphs containing two vertices u and v, a walk from u to v negatively signed and another walk from u to v positively signed.

Incidently, notice that if $a = (u, v) \in A$ is an arc such that the edge (u, v) of the underlying undirected graph does not belong to any cycle of this graph, then the fact that G is a possible signed digraph or an impossible one is independant of $sign_G(a)$.

Algorithm 1 is adapted from the famous algorithm [7] gives a topological order on a digraph without circuits. Given a possible signed digraph H and its reduced digraph $G = \mathcal{RD}(H)$, algorithm 1 works on the graph without prohibited cicuits, $\mathcal{RO}(G)$. It returns in time $\mathcal{O}(|V| + |A|)$ an update schedule s such that $\mathcal{NS}(G)^s = G$.

Remark 4.7 *The update schedule s returned by this algorithm is such that*

$$max\{s(v) \mid v \in V\} = min\{max\{s'(v) \mid v \in V\} \mid s' \in \mathcal{I}(G)\}.$$

Figure 4.1 shows the different steps of the algorithm that returns an update schedule associated to an arbitrary possible signed digraph (not necessarily reduced).

In the end, the following result holds.

Theorem 4.8 *Let G be a signed digraph. Both of the following problems can be solved in polynomial time.*

1. *Determine whether G is possible or not,*
2. *Find an update schedule s such that $(\mathcal{NS}(G))^s = G$.*

Preuve According to theorem 4.6, a signed digraph G is possible if and only if, in $\mathcal{RO}(G)$ no $>$-arc belongs to a strongly connected component. Thus, the first part of theorem 4.8 holds since the strongly connected components of a digraph can be identified in polynomial time.

The second part of theorem 4.6 comes from the existence algorithm 1 whose run time is also polynomial. $\square$

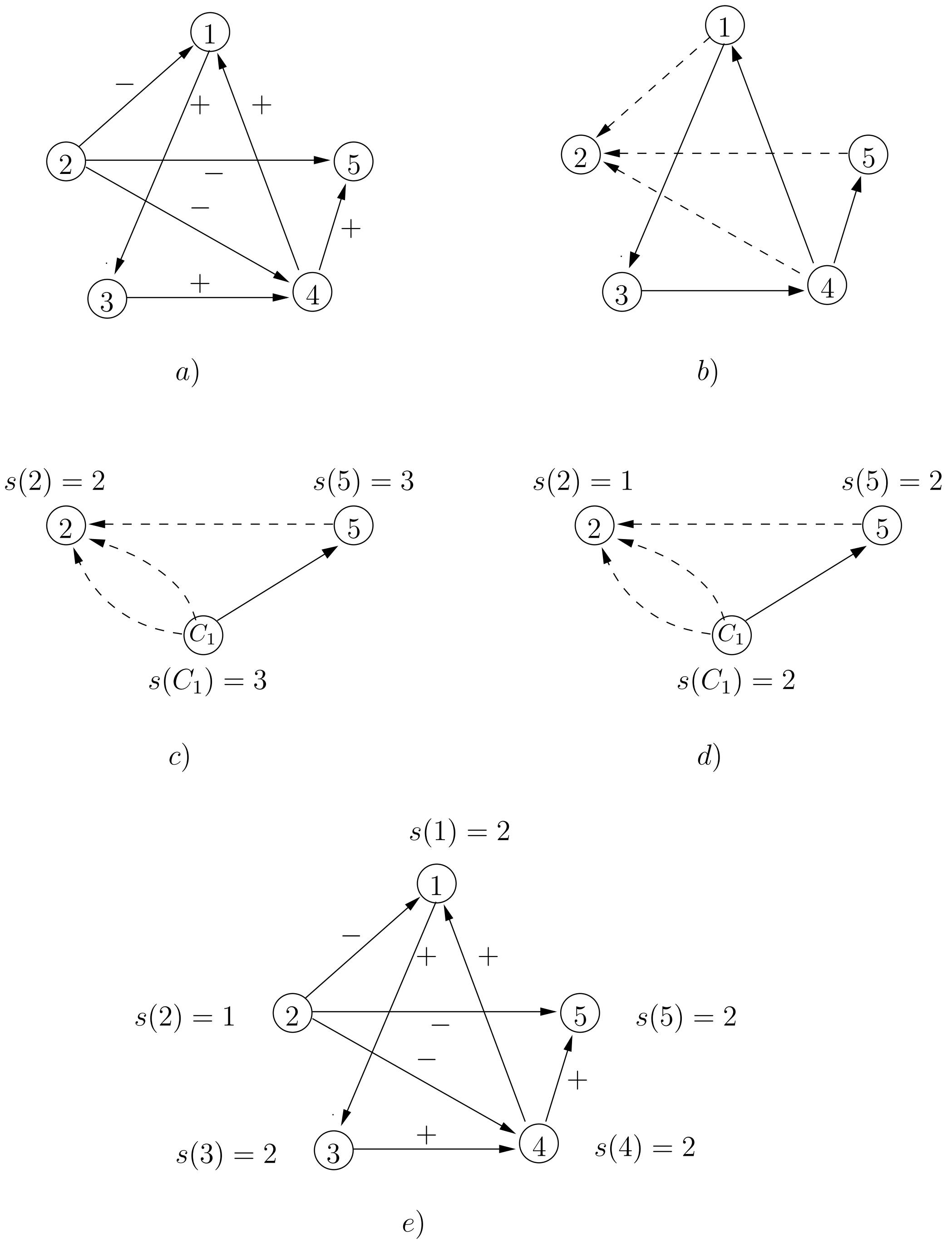


Figure 3: *a*. A signed digraph $G = (\{1, \ldots, 5\}, A)$. *b*. $\mathcal{RO}(G)$. The arcs drawn in dotted lines are $>$-arcs. The others are $\geq$-arcs. *c*. $H = \mathcal{RD}(\mathcal{RO}(G)) = \mathcal{RO}(\mathcal{RD}(G))$ and the update schedule computed by algorithm 1 after the *while loop*. $C_1 = \{1, 3, 4\}$. *d*. The digraph H and the update schedule returned by algorithm 1 with H as input. *e*. The update schedule s such that $G = (\mathcal{NS}(G))^s$.

Algorithm 1: `update schedule associated to an signed digraph`

```
Input: G = (V, A) reduced signed digraph, reoriented and sans circuit interdit
begin
    ValMax ← table of size |V(G)| in which are stocked the maximal possible
    values of s(v), v ∈ V(G). n ← |V|;
    H ← G;
    forall v ∈ V do
        ValMax[v] = n;
    end
    while ∃v ∈ V, deg⁻(v) = 0 do
        s(v) ← ValMax[v];
        forall u ∈ N(v) do
            if (u, v) is a >-arc then
                ValMax[u] ← min{ValMax[u], s(v) − 1};
            else
                ValMax[u] ← min{ValMax[u], s(v)};
            end
            delete the arc (u, v) from H;
        end
    end
    s_min ← min{s(v | v ∈ V)};
    forall v ∈ V do
        s(v) ← s(v) − s_min;
    end
end
```

4.2 Equivalence classes $[\cdot]^{GS}$

In this section we study the equivalence classes $[\cdot]^{GS}$. Finding the size of one these classes is the same as finding the number of partial orders (not necessarily strict) that we can associate to a reoriented signed digraph, respecting the « meaning » of its $>$-arcs and $\geq$-arcs. We are not yet able to say wether or not there exists a formula giving the size of a class, neither wether it is possible or not to determine in polynomial time if a given class $[s]^{GS}$ is such $|[s]^{GS}| > k, k \in \mathbb{N}$. However, we have the following proposition. Although it does not answer these questions, it is about a problem that is related to them.

Proposition 4.9 *The following problem which we chose to call* $\sharp\mathrm{O}A$ *is* $\sharp P$*-complete*[2]*.*

Input:	*A signed digraph* $G = (V, A)$*;*
Question:	*How many update schedules* s *are there such that* $G = (\mathcal{NS}(G))^s$ *and the number of arcs* $a = (u, v) \in A$ *satisfying* $sign_G(u, v) = +$ *and* $s(u) > s(v)$ *is minimum ?*

[2]See [2] for a definition of the complexity class $\sharp$P.

Two other versions of problem $\sharp O$A as well as the proof of proposition 4.9 are given in annex B.

Let us now consider the following question : given a digraph G and an update schedule s, does any udpdate schedule $s' \neq s$ such that $G^s = G^{s'}$ exist? That is, what conditions need to be satisfied in order for $|[s]^{GS}| > 1$ to hold?

Lemma 4.10 *Let $G = (V, A)$ be a reduced possible signed digraph. Let L be the length (counting the number of vertices) of the longest walk in $\mathcal{RO}(G)$. Then $\forall m \in [\![L, |V|]\!]$, there exists an update schedule s such that $max\{s(v) \mid v \in V\} = m$ and $G = (\mathcal{NS}(G))^s$.*

Preuve Let $\mathcal{RO}(G) = H = (V_H, A_H)$. Also, let $L(v)$ designate the length (counting the number of vertices) of the longest walk in H from a vertice u such that $deg^-(u) = 0$ to the vertice v. In particular, if $deg^-(v) = 0$ then $L(v) = 1$. Let us first suppose that $\forall a \in A_H,\ sign_H(a) = -$. We show the result in this specific case by induction on m.

If $m = L$, then the following function suits our purposes :

$$s : \left\{ \begin{array}{ccc} V_H & \longrightarrow & [\![1, |V_H|]\!] \\ v & \longmapsto & L - L(v) + 1 \end{array} \right.$$

Indeed, let $(v_1, \ldots, v_L)$ be a walk of H of length L ($\forall i < L,\ (v_i, v_{i+1}) \in A_H$). Then, since necessarily $deg^-(v_1) = 0$, $s(v_1) = L$ holds and for any other integer $k < L$ there exists i such that $s(v_i) = k$. Thus, s is indeed an update schedule such that $max\{s(v)|\ v \in V\} = L$. On the other hand, let $a = (u, v) \in A_H$ and $w \in V_H$ such that there exists a walk in H of length $L(u)$ from w to u. Then the walk $(w, \ldots, u, v)$ is a walk of length $L(u) + 1 > L(u)$ so $s(v) = L - L(v) + 1 \leq L - (L(u) + 1) + 1 < s(u) = L - L(u) + 1$. The update schedule s does satisfy $G^s = G$.

Let $m \in [\![L, |V|[\![$. Suppose that there exists an update schedule s such that $G^s = G$ and $max\{s(v) \mid v \in V\} = m$. Since $m < |V|$, there exists $i \in [\![1, m]\!]$ such that $|B_i^s| > 1$. Let $v \in B_i$. The update schedule s' défined in the following way cleary satisfies the desired properties :

$$s'(u) = \begin{cases} s(u) + 1 & \text{if } u \text{ belongs to a walk from a vertice } w \text{ such that } deg^-(w) = 0 \text{ to } v \\ s(u) & \text{otherwise} \end{cases}$$

where a walk can be of length 0, *i.e.,* $s'(v) = s(v) + 1$.

Now, suppose that the digraph H contains $\geq$-arcs. Then obviously, any update schedule s such that $\forall a = (u, v) \in A_H,\ s(u) > s(v)$ is suitable and the result in this case can be infered from the preceeding proof. □

Corollary 4.11 *Let $G = (V, A)$ be possible signed reduced digraph and L the length (counting the number of vertices) of the longest walk in $\mathcal{RO}(G)$. Then $|\mathcal{I}(G)| \geq |V| - L + 1$.*

Remark 4.12 *There exists two cases in which $|\mathcal{I}(G)| = 1$:*

1. *If G is linear digraph then $L = |V|$. Thus, in the case where $\forall a \in A$, $sign_G(a) = -$, only one update schedule s satisfies $G = (\mathcal{NS}(G))^s$. On the contrary, if $\exists a \in A$, $sign_G(a) = +$, there are 2^k such update schedules, where k is the number positive arcs of G.*

2. *If G is strongly connected, $\mathcal{RD}(G)$ is reduced to one lone vertice. According to the previous lemma, only one schedule satisfies $G = (\mathcal{NS}(G))^s$: the parallel update schedule, s_p.*

Theorem 4.13 *Let $G = (V, A)$ be a possible signed digraph. $|\mathcal{I}(G)| > 1$ if and only if G is neither such that $\mathcal{RO}(G)$ is strongly connected, neither such that $\mathcal{RD}(G)$ is linear and negative.*

Preuve If G is neither such that $\mathcal{RO}(G)$ is strongly connected, neither such that $\mathcal{RD}(G)$ is linear and negative, then $|[\![L, |V(\mathcal{RD}(G))|]\!]| > 1$. Thus, by lemma 4.10, $|\mathcal{I}(G)| > 1$. Reversely, if G is of one of these two types of signed digraphs, then by the remark 4.12, $|\mathcal{I}(G)| = 1$. □

Finally, because G^{s_p} cannot be a negative linear digraph (G^{s_p} is always a positive signed digraph) and because for any update schedule s, $\mathcal{I}(G^s) = [s]^{GS}$, the following corollary holds :

Corollary 4.14 *Let G be digraph. $|[s_p]^{GS}| > 1$ if and only if G is not strongly connected.*

4.3 Digraph signatures

In the previous section, we were given a signed digraph G and we were concerned by the existence of an update schedule of the vertices of G, s, such that $(\mathcal{NS}(G))^s = G$. And, in the case there did exist one we wanted to know how many there were. Here, we are given an unsigned digraph and we would like to determine which are the possible signatures of this digraph. In other words, it is in the *number* of equivalence classes $[\cdot]^{GS}$ that we are interested, rather than in their *sizes*.

We do not yet know whether the problem consisting in deciding if a given digraph can be signed by at least k different possible signatures (k being an integer) is NP-complete or polynomial. In annex A figures an algorithm that takes as input a digraph $G = (V, A)$ and enumerates the possible signed digraphs associated to G. The complexity of this algorithm is *A priori* exponential. The algorithm works by trial and error. It tries all $2^{|A|}$ signatures of the digraph G but skips a signature and moves on to the next as soon as it realizes that it is building an impossible signature because it is about to close a prohibited circuit in the reoriented graph, The problem of knowing at which moment the algorithm how many times the algorithm has to skip an impossible signature is a problem that is very close to the central questions of this section, that is, what are the properties of the digraph that the possible signatures are related to and how many of these are there. The results and remarks 4 to 4.17 of this section came from our search of an answer to these questions.

Proposition 4.15 *Let $G = (V, A)$ be a digraph and B a set of arcs of A such that there exists no circuit in G whose arcs all belong to B. Then, there exists an update schedule such that $\forall a \in B,\ sign_s(a) = -$.*

Preuve Let $k = |B|$ be an arbitrary integer. We prove lemma 4 by induction on $m = |A|$.

If $m = k$, by hypothesis, G does not contain any circuit. So if s is the update schedule such that $\forall a \in A = B,\ sign_s(a) = -$, $\mathcal{RO}(G^s)$ does not contain any circuits either and by theorem 4.6, G^s is possible.

Suppose that the property to be proven is true for all digraphs G such that $|A(G)| = m \geq k$. Let $G = (V, A)$ be a digraph such that $|A| = m + 1$ and $B \subseteq A$ a set of arcs of size $|B| = k$ such that there does not exist any circuit in G whose arcs are arcs of B. Let a_0 be an arc of $A \setminus B$. Define $G' = (V, A')$ where $A' = A \setminus \{a_0\}$. By hypothesis of induction, there exists an update schedule s' such that $\forall a \in B,\ sign_{s'}(a) = -$.

Let σ be a signature of the arcs of G such that $\forall a \in A \setminus \{a_0\}$, $\sigma(a) = sign_{s'}(a)$. We define $\sigma(a_0)$ such that the digraph G signed by σ is possible, *i.e.* such that there exists an update schedule s satisfying $sign_s = \sigma$. If, by adding the arc a_0 in $\mathcal{RO}(\ G'^{s'})$, no prohibited circuit is closed, then, a_0 can be signed by a $+$. Otherwise, suppose that $a_0 = (u, v)$. Let $C = (v = v_1, \ldots, v_l = u)$ be a walk of $\mathcal{RO}(G'^{s'})$ containing a $>$-arc. If $\sigma(a_0) = +$, then $C \cup \{a_0\}$ is a prohibited circuit but if $\sigma(a_0) = -$, the signature σ becomes possible : the digraph G signed by σ is then without any prohibited circuit. Indeed, on one hand, the arc $a_0 = (u, v)$ having changed direction, $C \cup \{(v, u)\}$ no longer is a circuit in this digraph. And, on the other hand, if there exists another walk $C' = (u = w_1, \ldots, w_k = v)$ closed by the arc (v, u), then $C \cup C' = (v = v_1, \ldots, v_l = u = w_1, \ldots, w_k = v)$ (see figure 4.3) must be a prohibited circuit of $\mathcal{RO}(G'^{s'})$, which is a contradiction. □

Corollary 4.16 *Let $G = (V, A)$ be a digraph. For every arc $a \in A$, there exists a way to sign the arcs of G to obtain a possible signed digraph $\widetilde{G}$ such that $sign_{\widetilde{G}}(a) = -$.*

Remark 4.17 *Let $G = (V, A)$ be a digraph whose non oriented underlying graph is a cycle. Two different cases are possible :*

1. *G is a circuit. In this case, the only impossible signed digraph $\widetilde{G}$ associated to G is the signed digraph such that $\forall a \in A,\ sign_{\widetilde{G}} = -$. The other $2^{|A|} - 1$ signed digraphs associated to G are therfore possible;*

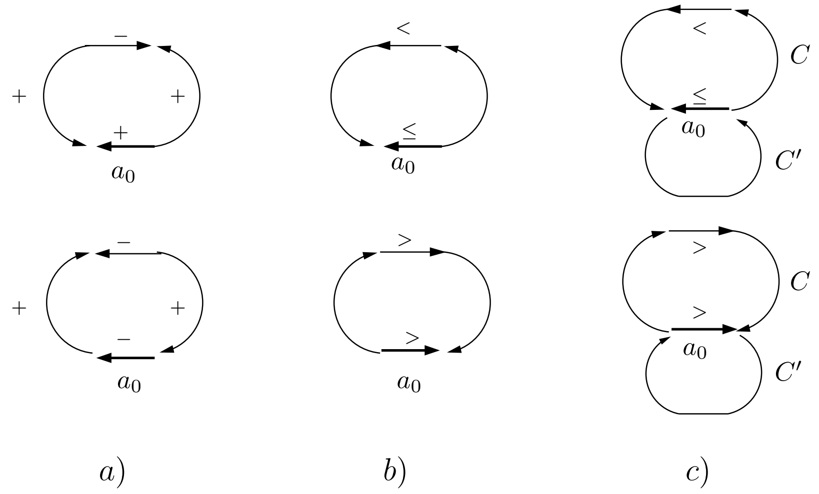


Figure 4: *a.* The arc a_0 and a «walk» C in the associated signed digraph. In these two examples, only one arc of the walk is signed negatively. *b.* The same arcs in the reoriented digraph. a_0 has been signed in way such that C is not a prohibited circuit. *c.* If there exists another «walk» C' closed by the arc a_0, then we obtain a contradiction : $C \cup C'$ is a prohibited circuit.

2. G is not a circuit. Then there exists two impossible signed digraphs associated to G : one for each possible orientation of a prohibited circuit of the reoriented digraph.

Finally, as it was the case for the number of update schedules associated to a given signed digraph, that is the sizes of the $[\cdot]^{GS}$ classes , we are not yet able to say whether there exists a formula expressing the number of possible signatures of a digraph, *i.e.*, the number of $[\cdot]^{GS}$ classes. Neither can we yet say whether there exists or not a polynomial algorithm that enumerates those classes. To end this section, let us now give the following proposition 4.18 which is prooved in annex C. Although it does not answer these questions, it concerns a problem that is however quite close to them.

Proposition 4.18 *The following problem, let us call it* $\sharp$GS*, is* $\sharp P$*-complete.*

Input: *A couple* (G, k) *where* $G = (V, A)$ *is a digraph and* $k \leq |V|$ *is an integer ;*

Question: *What is the number of signed digraphs* G^s *associated to* G *such that there exists a set* $W \subseteq V$ *satisfying :*

$*$ $|W| = k$ *and*

$*$ $a = (v_1, v_2) \in A \cap (W \times V) \Leftrightarrow sign_{G^s}(a) = -$ *?*

In other words, the question is “How many ways are there to correctly sign the digraph G such that there exists a set W of size exactly k containing all vertices that have an outgoing negative arc as well as, possibly, some vertces with a null out-degree?”

5 On the dynamics of boolean networks

As mentioned above, in addition to the the signed digraph related questions, we also have been looking to make comparisons between pairs of boolean networks whose dynamics are identical without the signature of their associated digraphs being the same. In section 5.1, we start by suggesting a way to construct two networks R_1 and R_2 such that $R_1 \overset{D}{\sim} R_2$ and $\neg(R_1 \overset{GS}{\sim} R_2)$. Then, in sections 5.2 eand 5.3, we study two simple cases : that of networks whose associated digraphs are circuits, and that of networks whose associated digraphs are complete digraphs without loops.

5.1 Construction of boolean networks with identical dynamics

In section 3 we have seen that there exists networks R_1 and R_2 who present the same dynamical behavior but different signed digraphs. Figure 5.1 shows a way to build two such networks of arbitrary sizes and associated to the same digraph. The two networks are in fact identical except for the images of two vertices by the update schedule. In this example, the update schedules of both networks, respectively s_1 and s_2, are sequential and for almost every vertice $i \in V$, $N^+_{s_k}(i) = \emptyset$ $(k \in \{1,2\})$. Proposition 5.1 following figure 5.1 is a generalisation of this construction to networks whose update schedules are not necessarily sequential.

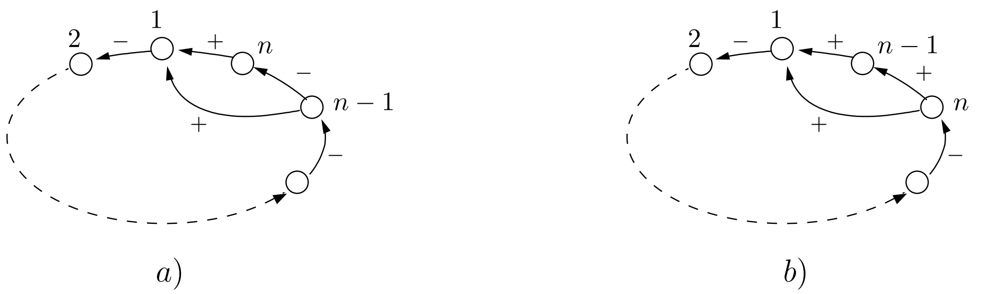


Figure 5: Two networks $R_i = (G, F, s_i)$, $i \in \{1,2\}$ whose signed digraphs are différent. The numbers near the vertices indicate the image of the vertices by the update schedule s_i, $i \in \{1,2\}$.

Proposition 5.1 *Let $R_1 = (G, F, s_1)$ and $R_2 = (G, F, s_2)$ be two boolean networks with the same underlying digraph $G = (V, A)$ without loops and such that*

1. *The global activation function F is defined by local activation functions that are all identical, symmetrical, and just as the OR and AND functions, they also are associative, i.e., $\forall i \in V$, $\forall V_1, V_2 \subseteq V$,*

$$\begin{aligned} f_i(x_j \mid j \in V_1 \; ; \; y_j \mid j \in V_2) &= f_i(f_i(x_j \mid j \in V_1) \; ; \; y_j \mid j \in V_2) \\ &= f_i(x_j \mid j \in V_1 \; ; \; f_i(y_j \mid j \in V_2)) \\ &= f_i(f_i(x_j \mid j \in V_1) \; ; \; f_i(y_j \mid j \in V_2)) \end{aligned}$$

and if $V_1 \subseteq V_2$ *then* $f_i(x_j \mid j \in V_1 \ ; \ x_j \ \mid j \in V_2) = f_i(x_j \mid j \in V_2)$,

2. $B_1^{s_1} = \{1\}$[3],
3. $\forall i \in V,\ N^+_{s_1}(i) \subseteq N(1)$
4. *If* $n = max\{s(i) \mid i \in V\}$, $B^{s_1}_{n-1} \subseteq N(1)$ *and* $B^{s_1}_n \subseteq N(1)$,
5. *And* s_2 *is such that*

$$B_t^{s_2} = \begin{cases} B_t^{s_1} & \text{if } t < n-1 \\ B_n^{s_1} & \text{if } t = n-1 \\ B_{n-1}^{s_1} & \text{if } t = n \end{cases}$$

Then, $G^{s_1} \neq G^{s_2}$ *and* $F^{s_1} = F^{s_2}$.

Preuve Suppose the vertice $1 \in V$ is such that $s_1(1) = s_2(1) = 1$ and that $\forall i \in V$, $f_i = f$. Note that $N^-_{s_1}(1) = N^-_{s_2}(1) = \emptyset$.

Consider the network R_1. By induction on t, we prove that

$$\forall x \in \{0,1\}^n,\ \forall i \in B_t^{s_1},\ f_i^{s_1}(x) = f_1^{s_1}(x).$$

If $t = 2$, $\forall i \in B_2^{s_1}$ the following holds thanks to the hypotheses on f and because $N^-_{s_1}(i) \subseteq \{1\}$:

$$\begin{aligned} f_i^{s_1}(x) &= f(x_j \mid j \in N^+_{s_1}(i) \ ; \ f_j^{s_1}(x) \mid j \in N^-_{s_1}(i)) \\ &= f(x_j \mid j \in N^+_{s_1}(i) \cap N(1) \ ; \ f_1^{s_1}(x)) \\ &= f(x_j \mid j \in N^+_{s_1}(i) \cap N(1) \ ; \ f(x_j \mid j \in N(1))) \\ &= f(x_j \mid j \in N(1)) \\ &= f_1^{s_1}(x) \end{aligned}$$

Suppose that $t \geq 2$ and that $\forall i$, $s_1(i) < t$, $f_i^{s_1}(x) = f_1^{s_1}(x)$. Then, in the same way it holds that $\forall i \in B_t^{s_1}$,

$$\begin{aligned} f_i^{s_1}(x) &= f(x_j \mid j \in N^+_{s_1}(i) \ ; \ f_j^{s_1}(x) \mid j \in N^-_{s_1}(i)) \\ &= f(x_j \mid j \in N^+_{s_1}(i) \cap N(1) \ ; \ f_1^{s_1}(x)) \\ &= f_1^{s_1}(x). \end{aligned}$$

Now, consider the network R_2. $\forall t < n-1$, $\forall i \in B_t^{s_1} = B_t^{s_2}$, $f_i^{s_2}(x) = f_i^{s_1}(x) = f_1^{s_1}(x)$. Also, because $\forall i \in B^{s_2}_{n-1}$, $N^+_{s_2}(i) \subseteq B^{s_2}_{n-1} \cup B^{s_2}_n = B^{s_1}_n \cup B^{s_1}_{n-1} \subseteq N(1)$ and $j \in N^-_{s_2}(i) \Rightarrow j \in B_t^{s_2},\ t < n-1$, the following holds:

$$\begin{aligned} \forall i \in B^{s_2}_{n-1}, \quad f_i^{s_2}(x) &= f(x_j \mid j \in N^+_{s_2}(i) \ ; \ f_j^{s_2}(x) \mid j \in N^-_{s_2}(i)) \\ &= f(x_j \mid j \in N^+_{s_2}(i) \cap N(1) \ ; \ f_1^{s_1}(x)) \\ &= f_1^{s_1}(x). \end{aligned}$$

[3] Recall that $B_t^s = \{i \in V \mid s(i) = t\}$ (*cf* notation 2.3).

Also, $\forall i \in B_n^{s_2} = B_{n-1}^{s_1}$,

$$N_{s_2}^{+}(v) \subseteq B_n^{s_2} = B_{n-1}^{s_1} \subseteq N(1) \quad \text{and} \quad N_{s_2}^{-}(i) \subseteq N_{s_1}^{-}(i) \cup B_n^{s_1} = N_{s_1}^{-}(i) \cup B_{n-1}^{s_2}$$

and for every vertice $j \in N_{s_1}^{-}(i) \cup B_{n-1}^{s_2}$, $j \in B_t^{s_2}$, $t \leq n-1$. Thus, $f_j^{s_2}(x) = f_1^{s_1}(x)$ and consequently,

$$\begin{aligned} \forall i \in B_n^{s_2}, \quad f_i^{s_2}(x) &= f(x_j \mid j \in N_{s_2}^{+}(i)\ ;\ f_j^{s_2}(x) \mid j \in N_{s_2}^{-}(i)) \\ &= f(x_j \mid j \in N_{s_2}^{+}(i) \cap N(1)\ ;\ f_1^{s_1}(x)) \\ &= f_1^{s_1}(x). \end{aligned}$$

□

5.2 The dynamics of circuits

Let us now look at the case of boolean networks whose underlying digraphs are circuits. In the sequel, we write $C_n = [v_0, v_1, \ldots, v_{n-1}, v_n = v_0]$ to denote the circuit of size n *i.e.*, the digraph of order n whose vertices are $v_1, \ldots, v_{n-1}, v_n = v_0$, and whose set of arcs is $\{(v_i, v_{i+1}) \mid i \in [\![0, n[\![\}$.

Consider the boolean network $R = (C_n, F, s)$. f_i designates the local activation function of a vertice v_i and x_i denotes its state. If all local activation functions are equal to the same function $f : \{0,1\} \rightarrow \{0,1\}$ which is non constant, $\forall v_i \in V$ the following holds :

$$f_i^s(x) = \begin{cases} x_{i-1} & \text{if } sign_s(v_{i-1}, v_i) = +1 \\ f_{i-1}^s(x) & \text{if } sign_s(v_{i-1}, v_i) = -1 \end{cases}$$

if f is the identity function and

$$f_i^s(x) = \begin{cases} \neg x_{i-1} & \text{if } sign_s(v_{i-1}, v_i) = +1 \\ \neg f_{i-1}^s(x) & \text{if } sign_s(v_{i-1}, v_i) = -1. \end{cases}$$

if f is the negation function. Let $i_s = max\{j < i \mid sign_s(v_j, v_{j+1}) = +1\}$. By induction we can easily prove that in the first case $\forall v_i \in V$, $f_i^s(x) = x_{i_s}$ and in the second :

$$f_i^s(x) = \begin{cases} \neg x_{i_s - 1} & \text{if } i - i_s \text{ isimpair} \\ f_{i_s-1}^s(x) & \text{if } i - i_s \text{ ispair.} \end{cases}$$

Proposition 5.2 *Let $R_1 = (C_n, F, s_1)$ and $R_2 = (C_n, F, s_2)$ be two boolean networks such that $C_n^{s_1} \neq C_n^{s_2}$ and such that F is defined by one unique non constant local activation function $f : \{0,1\} \rightarrow \{0,1\}$. Then $F^{s_1} \neq F^{s_2}$. In other words, $C_n^{s_1} = C_n^{s_2} \Leftrightarrow F^{s_1} = F^{s_2}$.*

Preuve If s_1 and s_2 are two different update schedules such that $C_n^{s_1} \neq C_n^{s_2}$, there exists $v_i \in V$ such that $sign_{s_1}(v_{i-1}, v_i) = +$ and $sign_{s_2}(v_{i-1}, v_i) = -$ (or conversely).

Then if f is the identity function and $x \in \{0,1\}^n$ is a point such that $x_{i_{s_1}} = x_{i-1} = 1$ and $x_{i_{s_2}} = 0$, $F^{s_1}(x)_i = x_{i-1} = 1 \neq F^{s_2}(x)_i = x_{j_{s_2}} = 0$.

If f is the negation function, in a simmilar manner, we can prove that there exists a point $x \in \{0,1\}^n$ such that $F^{s_1}(x)_i \neq F^{s_2}(x)_i$. $\square$

5.3 The dynamics of complete digraphs

Here we study boolean networks whose underlying digraphs are complete digraphs without loops of order n and whose local activation functions are symmetrical and all equal to the same function OR, AND, or any other commutative function $\odot$ such that $\forall a, b \in \{0,1\}$, $(a \odot b) \odot a = a \odot b$. We will, however, chose to take the OR function here as an example and we will write it $\vee$. The goal of this section is to prove theorem 5.5. This theorem states that the dynamics of a boolean network $R = (G, \mathrm{OR}, s)$ associated to a complete digraph without loops depends on the block B_1^s.

Notation 5.3 *In order to simplify notations, we will write $f_i^s(x) = x'_i$ when there will be no ambiguity as to what update schedule s is being considered.*

Lemma 5.4 *Let $R = (G, F, s)$ be a boolean network associated to a complete digraph without loops $G = (V, A)$ of order n.*

1. *If s is parallel ($B_1^s = V(G)$), $\forall i \in V$,*

$$x'_i = \bigvee_{j \neq i} x_j,$$

2. *If s is sequential or block sequential such that $B_1^s = \{i^*\}$, $\forall i \in V$,*

$$x'_i = \bigvee_{j \neq i^*} x_j,$$

3. *If s is block sequential and $|B_1^s| > 1$,*

$$\forall i \in B_1^s,\ x'_i = \bigvee_{j \neq i} x_j, \qquad \forall i \notin B_1^s,\ x'_i = \bigvee_{j} x_j.$$

Preuve

1. Trivial.

2. Clearly, the following holds

$$x'_{i^*} = \bigvee_{j \neq i^*} x_j$$

and $\forall i \in B^s_t$, $t > 1$, (by induction on t) we have that

$$x'_i = [\bigvee_{s(j)<t} x'_j\ ] \vee [\bigvee_{s(j)\geq t} x_j\ ] = [\bigvee_{j\neq i^*} x_j\ ] \vee [\bigvee_{s(j)\geq t} x_j\ ] = \bigvee_{j\neq i^*} x_j.$$

3. The first equality is trivial. The second comes from an induction and from, in particular,

$$\forall i \in B^s_2, \qquad x'_i = [\bigvee_{s(j)\geq 2} x_j\ ] \vee [\bigvee_{j\in B^s_1} x'_j\ ] = \bigvee_j x_j.$$

□

Let s_1 and s_2 be two different update schedules such that $G^{s_1} \neq G^{s_2}$. If one of these two update schedules is parallel and the other sequential or block sequential, it is clear that there exists $x \in \{0,1\}^n$ such that $F^{s_1}(x) \neq F^{s_2}(x)$. The same is true if both schedules are sequential ou block sequential such that $|B^{s_1}_1| = |B^{s_2}_1| = 1$ and $B^{s_1}_1 \neq B^{s_2}_1$.

Now suppose that s_1 is sequential or block sequential such that $B^{s_1}_1 = \{i^*\}$, and s_2 is block sequential. Si $\exists k \neq i^*,\ k \in B^{s_2}_1$, then

$$\forall x \in \{0,1\}^n, \quad F^{s_1}(x)_k = [\bigvee_{j\neq i^*, j\neq k} x_j\ ] \vee\ x_k \qquad \text{and} \qquad F^{s_2}(x)_k = [\bigvee_{j\neq i^*, j\neq k} x_j\ ] \vee\ x_{i^*}.$$

Considering for example a point $x \in \{0,1\}^n$ such that $\forall i \neq k,\ x_i = 0$ and $x_k = 1$, we can infer that $R_1 = (G, F, s_1)$ and $R_2 = (G, F, s_2)$ cannot have the same dynamics unless $B^{s_2}_1 = \{i^*\}$.

Finally, suppose that s_1 and s_2 are both block sequential update schedules such that $B^{s_1}_1 > 1$, $B^{s_2}_1 > 1$ and $B^{s_1}_1 \neq B^{s_2}_1$. Then if $i \in B^{s_1}_1 \setminus B^{s_2}_1$, necessarily there exists $x \in \{0,1\}^n$ such that

$$F^{s_1}(x)_i = \bigvee_{j\neq i} x_j \neq F^{s_2}(x)_i = \bigvee_j x_j.$$

From all the previous remarks we infer one of the directions of the following theorem. The other direction is clear by lemma 5.4.

Theorem 5.5 *Let $R_1 = (G, F, s_1)$ and $R_2 = (G, F, s_2)$ be two boolean networks associated to the same complete digraph without loops and such that $F = OR$ or any other one of the functions satisfying the properties mentioned above. Then, if $G^{s_1} \neq G^{s_2}$,*

$$F^{s_1} = F^{s_2} \Leftrightarrow B^{s_1}_1 = B^{s_2}_1.$$

In particular, the parallel update schedule cannot induce the same dynamical behavior as that induced by any other update schedule : $[s_p]^D = \{s_p\}$. Also, since there are 2^n ways of choosing the set B^s_1, the number of $[\cdot]^D$ classes for boolean networks whose associated digraph is a complete digraph without loops of order n is 2^n.

6 Conclusion

Thanks to theorem 3.1 of [1], we know that instead of giving an update schedule in the definition of a boolean network, we can give a *possible* signature of the digraph associated to the network. The reason why we are concerned by this notion of possible signature is that it is closely related to the notion of robustness of the dynamics of boolean networks with respect to update schedules : the smaller the number of equivalence classes $[\cdot]^{GS}$, *i.e,* the bigger the sizes of these classes, the more robust the network can be considered. All in all, this internship has mainly served to gain a better understanding of possible signatures, in particular by giving a characterisation of them. In this account, we have also showed that it is possible to check in polynomial time whether a signed digraph is possible or not. If it is, there exists a polynomial algorithm that returns an update schedule inducing the same signature of the underlying unsigned digraph. In addition, theorem 4.13 states that, subject to certain conditions, for any network R, there exists another network R' which is associated to the same digraph and which has the same dynamical behavior. Also, we have made some observations on the possible signatures of a given digraph and we have studied the cases of networks having similar dynamics but different signed digraphs. On one hand, we have showed how to build such networks and on the other hand we have compared dynamics of networks whose underlying digraphs are circuits or complete digraphs without loops.

Following the work we have done during this internship some questions remain to be answered. The main ones concern the complexity of the four problems given below as well as possible alternative wordings of them

- Given a digraph G, are there at least k different possible signatures of G?
- Given a signed digraph G^s or a digraph G and an update schedule s, are there at least k different update schedules s' such that $G^s = G^{s'}$?
- Given a boolean network $R = (G, F, s)$, are there at least k other boolean networks $R' = (G, F, s')$ (or even possibly $R' = (G', F', s')$) having the same dynamics as R?
- Given a digraph G and a global activation function F, are there at least k different boolean networks $R = (G, F, s)$ each having distinct dynamics?

Note that the second of these problems raises combinatoric questions and questions of order theory which are independant of boolean networks and of their dynamics. In particular, this problem is related to the number non strict partial orders that we can associate to a digraph and which, in addition, satisfy some special conditions (some arcs of the digraph must necessarily mean '>' whereas others can mean '>' as well as '=').

Acknowledgements

Finally, I would like to thank Lilian Salinas and Anahi Gajardo for their presence during this internship and Eric Goles for the interesting discussions we have had. I would also particularily like to express my sincere gratitude towards Julio Aracena for his great support and follow-up and for his availability.

References

[1] J. Aracena, E. Goles, and L. Salinas. On the robustness of update schedules in boolean networks. *Biosystems (submitted)*, 2008.

[2] S. Arora and B. Barak. *Computational Complexity: A Modern Approach*. Cambridge University Press, 2009.

[3] J. Bang-Jensen and G. Gutin. *Digraphs: Theory, Algorithms and Applications*. Springer-Verlag, 1979.

[4] J. Demongeot, A. Elena, and S. Sené. Robustness in regulatory networks: a multi-disciplinary approach. *Acta Biotheoretica*, 56(1-2):27–49, 2008.

[5] M. Garey and D. Johnson. *Computers and intractability: a guide to the theory of NP-completeness*. W.H. Freeman, 1979.

[6] E. Goles and L. Salinas. Comparison between parallel and serial dynamics of boolean networks. *Theoretical Computer Science*, 396:247–253, 2008.

[7] J. van Leeuwen. *Handbook of theoretical computer science*. Elsevier, 1990.

A An algorithm enumerating the possible signatures of digraph

In the following algorithm, if v a vertice of a digraph $G = (V, A)$,

$$\begin{aligned} A^{+}(v) &= \{a = (v, w) \in A \mid w \in V \text{ and } \mathtt{SIGN}(a) = +\} \cup \\ & \quad \{a = (w, v) \in A \mid w \in V \text{ and } \mathtt{SIGN}(a) = -\} \end{aligned}$$

and

$$\begin{aligned} A^{-}(v) &= \{a = (w, v) \in A \mid w \in V \text{ and } \mathtt{SIGN}(a) = +\} \cup \\ & \quad \{a = (v, w) \in A \mid w \in V \text{ and } \mathtt{SIGN}(a) = -\}. \end{aligned}$$

We suppose here that the arcs of the inputted digraph $G = (V, A)$ are numbered from 1 to $|A|$. For an arc $a = (v, w) \in A$, $\mathtt{beg}(a) = v$ and $\mathtt{end}(a) = w$. Also, algorithm 2 uses the function $\sigma : \mathbb{N} \times \mathbb{N} \to \{-, +\}$ defined in the following mannar:

$$\sigma(k, i) = \begin{cases} - & \text{si } k/.(2^{m-i}) \equiv 0 \; mod \; 2 \\ + & \text{si } k/.(2^{m-i}) \equiv 1 \; mod \; 2 \end{cases}$$

where $a/.b$ the quotient of the euclidian division of a by b. Finally, algorithm 2, while visiting *in depth* the digraph G, fills in a table « DONE » such that $\forall a \in A$, DONE$[a]$ =NO if the arc a hasn't yet been visited, DONE$[a]$ =IP if a is « in process », *i.e.* a has been visited and the current arc can be reached by following a walk in G starting by arc a. And, DONE$[a]$ =YES if the all arcs b reachable from arc a have been seen and are such that DONE$[b]$ =YES.

Algorithm 2: `Enumeration of the possible signed digraphs associated to a given digraph`

```
Input: a digraph G = (V, A)
begin
    n ← |V|;
    m ← |A|;
    MAT ← matrix of size m × n;
    DONE ← table of size m;
    SIGN ← table of size m;
    forall a ∈ A do DONE[a] ←NO;
    for k = 0 à 2^m − 1 do
        for i = 1 à m do SIGN[a_i] ← σ(k, i);
        while ∃a ∈ A, DONE[a] =NO do
            a ← any arc of A such that DONE[a] =NO;
            ALGO-AUX(a);
            if ∀a ∈ A, DONE[a] =YES then
                return table SIGN;
                REGRESS(MAT,DONE);
                go on to next k;
            end
            if ∀a ∈ A, DONE[a] =NO (signature k is impossible) then
                go on to next k;
            end
        end
    end
end
```

Algorithm 3: « `REGRESS` »(`MAT`,`DONE`)

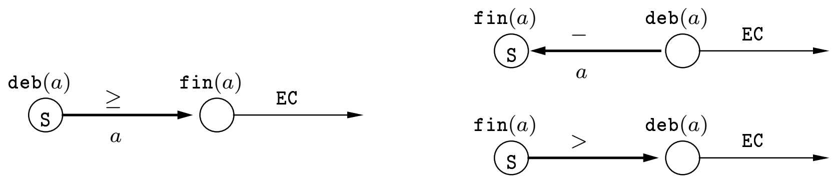
```
begin
    Empty table MAT;
    forall a ∈ A do DONE[a] ←NO
end
```

Algorithm 4: « `ALGO-AUX` »(a)

```
begin
    if SIGN[a] = − then
        v ← deb(a);
        if ∃b ∈ A⁺(v) such that DONE[b] = IP then
            REGRESS;
        else
            DONE[a] ← IP;
            MAT[a][v] ←S;
        end
    else
        v ← fin(a);
        if
          ( ∃b ∈ A⁺(v) such that DONE[b] =IP )  et  ( ∃b ∈ A, MAT[b][deb(a)] =S )
        then
            REGRESS;
        else
            DONE[a] ← IP;
            if ∃b ∈ A, MAT[b][deb(a)] =S then MAT[a][v] ←S;
        end
    end
    while ∃b ∈ A⁺(v) such that DONE[b] =NO do
        ALGO-AUX(b);
    end
    DONE[a] ← YES;
end
```

Proof of the signed digraphs enumeration algorithm

In algorithm 4, two calls to algorithm REGRESS are made. The first call coincides with the situation illustrated by figure A b. and the second one corresponds to the situation illustrated by figure A a.



$a)$ $b)$

Figure 6: a. Situation encountered during the run of ALGO-AUX when the second call to REGRESS is made. b. Situation encountered during the run of ALGO-AUX when the first call to REGRESS is made.

The letter S labelling a vertice v ($\exists a \in A, \mathtt{MAT}[a][v] = \mathtt{S}$) indicates that any arc leaving this vertice, *i.e.,* any arc of $A^+(v)$, belongs to a walk that contains a $>$-arc in the reoriented digraph corresponding to the signed digraph being built. Thus, it is clear that when the algorithm encounters one of the configurations represented in figure A it is building an impossible signature. Indeed, since the digraph is scanned in depth, if the algorithm finds an arc in the process of being dealt with (IP), it means that it has just covered a circuit of the corresponding reoriented digraph. In that case, if this circuit contains a $>$-arc, then the reoriented digraph contains a prohibited circuit. Therfore, we claim that any possible signature of the inputed digraph is effectively enumerated by the algorithm.

Conversely, no impossible signature is enumerated by the algorithm. More precisely, any impossible signature causes the algorithm to come upon one of the situations of figure A and to call algorithm REGRESS. Indeed, at any time, before it comes upon such a situation, no prohibited circuit of the reoriented digraph H corresponding to the signed digraph being built, has yet been seen. If C was such a circuit and $a \in A(H)$ the last arc of C dealt with by the algorithm, then,

- Either $C \setminus \{a\}$ would contain a $>$-arc b and this arc would be such that $\mathtt{DONE}(b) = \mathtt{IP}$ when a is seen for the first time (a being a descendant of b). But then the label S generated when b was being processed would have been passed along the walk from b to a and the algorithm would have encountered the situation of figure A a;

- Either $C \setminus \{a\}$ would contain only $\geq$-arcs and $\mathtt{SIGN}(a) = -$. But then, by definition of $a = (v, u)$, there would exist an arc $b \in A^+(v)$ belonging to the circuit C such that $\mathtt{DONE}(b) = \mathtt{IP}$ wich precisely is the situation of figure A b.

The following figure illustrates five steps of algorithm 2. The first one pictured is not the very first step of the algorithm ($k = 20$). In the caption of this figure, we improperly use the term *s*ignature since it is the reoriented digraphs that are represented (the arcs drawn in full lines are $\geq$-arcs and the arcs in dotted lines are $>$-arcs).

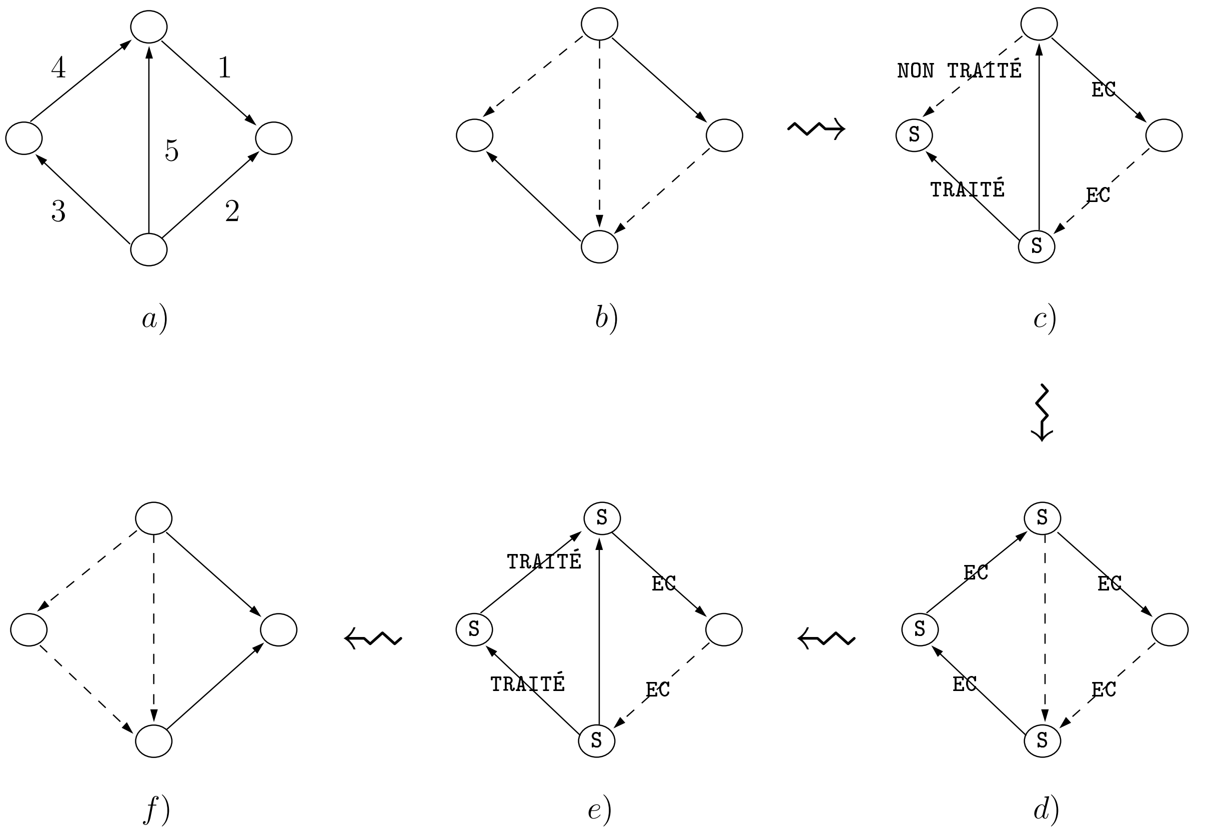


Figure 7: a) *The digraph G input of algorith 2.* The numbers are that of each arc. b) *The* 20^{th} *« signature » built during the run of algorith 2.* It is a possible signature. c) *The* 21^{st} *signature.* Here, we represent the « state of the digraph » when arc 5 is being dealt with.The situation corresponds to that of figure A a). The signature is impossible. d) *The* 22^{nd} *signature.* This figure pictures the state of the digraph when arc 5 is being dealt with. The present situation is similar to that pictured by figure A b). The signature, again, is impossible. e) *The* 23^{rd} *signature, impossible again (situation of figure A a)).* f) *The* 24^{th} *signature.* It is a possible signature.

B A problem relating to the sizes of the equivalence classes $[\cdot]^{GS}$

We prove proposition 4.9 of section 4.2 by reduction of the problem $\sharp C$ycle Cover. This problem consists in determining the number of spanning sub-digraphs of a digraph such that every vertice belongs to exactly one cycle.

Let $G = (V_G, A_G)$ be a digraph instance of $\sharp C$ycle Cover. Below, we show in a relatively informal manner how to transform G into a signed digraph $H = (V_H, A_H)$ which is an instance of $\sharp O$A.

1. To each arc $a = (u, v) \in A_G$ we associate four vertices, u, u_v, v_u and v, and six arcs amongst which the arc $a_u = (u, v_u)$ and the arc $a_v = (u_v, v)$ that both «represent» the arc a (*cf* figure 1);

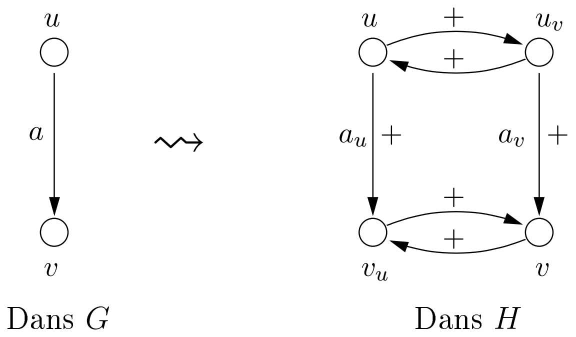


Figure 8:

2. For each vertice $v \in V$, we define the sets $A^-(v) = \{a_u = (u, v_u) \in A_H\}$ and $A^+(v) = \{a_u = (v_u, u) \in A_H\}$ and we arrange the arcs of both these sets in a way that they form a walk. In other words, we build a walk composed of the arcs representing ingoing arcs of v and another composed of the arcs representing outgoing arcs of v, for every vertice v of G. Doing so, some vertices created at step 1 merge;

3. For each of these walks we add a negative arc starting at the end of the walk and finishing at its beginning. In $\mathcal{RO}(H)$, these arcs being reversed, they become $>$-arcs that are oriented in the same direction as the walks.

Remark B.1 *The knowledge of an update schedule s such that $(\mathcal{NS}(H))^s = H$ is equivalent to the knowledge of the «meaning» of every $\geq$-arc of $\mathcal{RO}(H)$, that is, for every $a = (u, v)$, whether $s(u) > s(v)$ or $s(u) = s(v)$.*

Suppose that s is an update schedule such that $H = (\mathcal{NS}(H))^s$. Then, on one hand, s must satisfy

$$\forall a = (u, v) \in A_G, \quad s(u_v) = s(u) \text{ et } s(v) = s(v_u).$$

On the other hand, the existence of the negative arcs imposes that at least one of the $\geq$-arcs a_u of every set $A^-(v)$ (*resp.* $A^+(v)$) be such that $s(u) > s(v_u)$ (*resp.* $s(v_u) > s(u)$).

Notation B.2 *Given an update schedule s such that $(\mathcal{NS}(H))^s = H$, let us call D_H^s the following set of arcs of H : $\{a = (u,v) \in H \mid sign_H(u,v) = + \ et \ s(u) > s(v)\}$.*

Note that $a_u \in D_H^s \Leftrightarrow a_v \in D_H^s$.

If C_G is cycle cover of G, note that $C_H = \{a_u = (u, v_u),\ a_v = (u_v, u) \mid a = (u,v) \in C_G\}$. Define the update schedule s such that $C_H = D_H^s$ and for every other negative arc, $a = (u,v) \in A_H \setminus C_H$, $s(u) = s(v)$. Then s satisfies the properties required by problem $\sharp O$A. Indeed, on one hand, by construction, $H = (\mathcal{NS}(H))^s$. On the other, the set D_H^s is minimal because only one arc of each walk $A^-(v)$ or $A^+(v)$ belongs to C_H (otherwise, in C_G a vertice would be covered by more than one arc). Incidently, we have that $|D_H^s| = 2|C_G| = 4|V_G|$ (one ingoing arc and one outgoing arc for each vertice of V_G in the cycle cover, each of them being represented twice in C_H).

Conversely, if s is an update schedule such that $H = (\mathcal{NS}(H))^s$ and D_H^s is minimal (necessarily $|D_H^s| = 2|V_G|$), let us define the set $C_G = \{a = (u,v) \in A_G \mid \exists a_u \ et \ a_v \in C_H\}$. Then, since by minimality of D_H^s the following holds,

$$a_u = (u_v, v) \in C_H^s \quad \Rightarrow \quad \forall w \neq u,\ (w,v) \in A_G,\ a_w = (w_v, v) \notin C_H$$

$$a_u = (v, v_u) \in C_H^s \quad \Rightarrow \quad \forall w \neq u,\ (v,w) \in A_G,\ a_w = (v, v_w) \notin C_H^s,$$

no vertice of G is covered more than once. And because of the negative arcs, every one of them is covered at least once. Therefore, C_G is a cycle cover of G.

Remark B.3 *With the same reduction as the one just described, we can show that the following problems (very close to problem $\sharp$OA) also are $\sharp$P-complete.*

Input: *A signed digraph $G = (V, A)$ and a set $\{V_i \subseteq V \mid 1 \leq i \leq k\}$ of parts of V;*

Question: *What is the number of update schedules s such that $G = (\mathcal{NS}(G))^s$ and $\forall i \in [\![1, k]\!]$, there exists a unique arc $(u,v) \in A \cap (V_i \times V_i)$ satisfying $sign_G(u,v) = +$ and $s(u) > s(v)$?*

Input: *A signed digraph $G = (V, A)$ and an integer k ;*

Question: *What is the number of update schedules s such that $G = (\mathcal{NS}(G))^s$ and amongst the m positive arcs $a = (u,v)$ of G, there are exactly k of them such that $s(u) > s(v)$?*

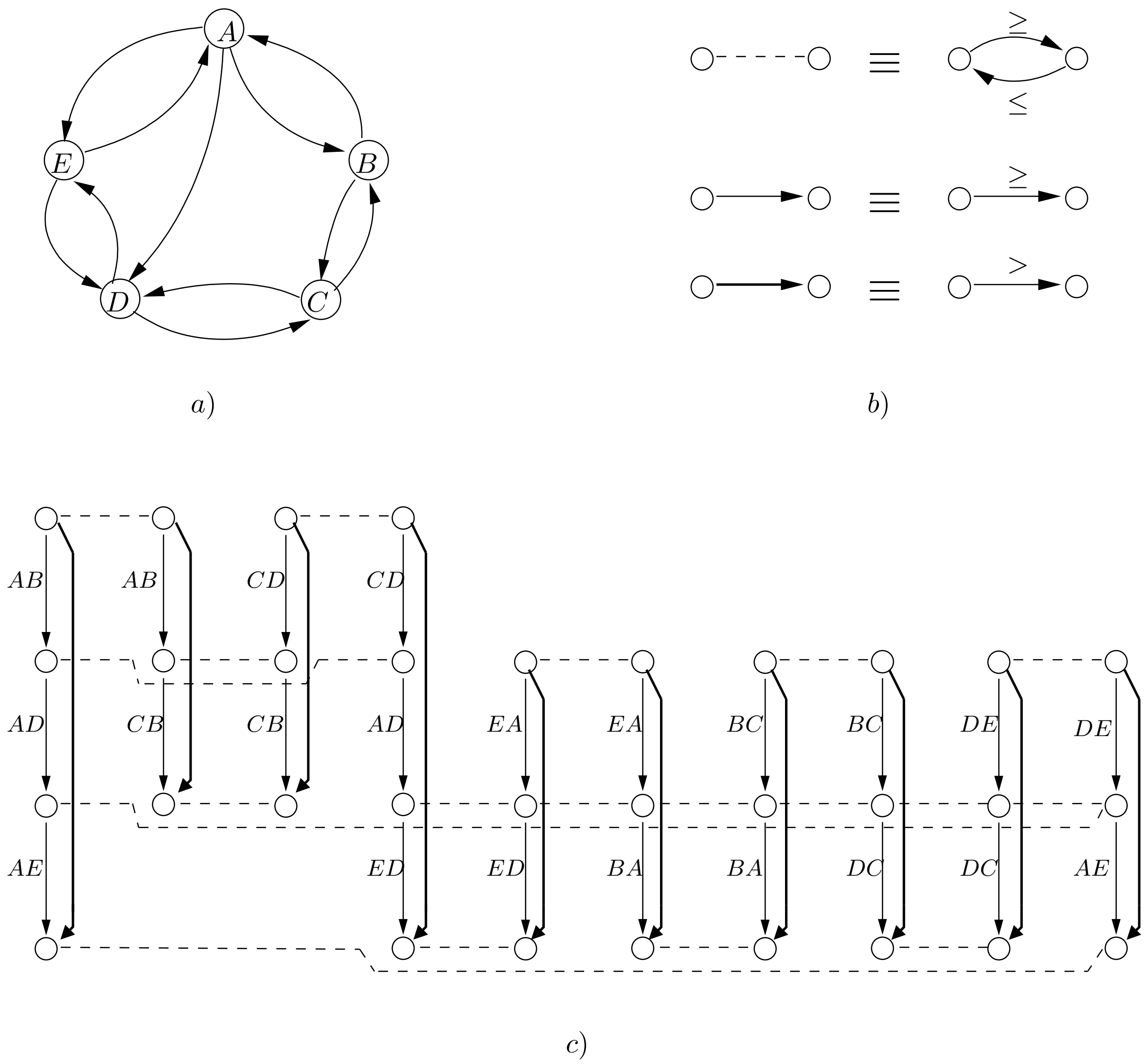


Figure 9: *a*. A digraph G, instance of $\sharp C$ycle Cover. *b*. Caption of figure *c*. *c*. Transformation of G into the digraph H, instance of $\sharp O$A. Here, we represent $\mathcal{RO}(H)$. In the intention of keeping the figure simple, we only have labelled the arcs representing an arc of G.

C A problem relating to the number of equivalence classes $[\cdot]^{GS}$

We prove proposition 4.18 by reduction of problem $\sharp P$erfect Matching for a bipartite graph ($\sharp P$M). This problem consists in finding the number of $p$erfect matchings of a bipartite graph[4].

Notation C.1 *Let $G = (V, A)$ be a signed digraph. We define $deg_+^+(v) = |\{(v,u) \in A \mid sign_G(v,u) = +1\}|$ and $deg_+^-(v) = |\{(u,v) \in A \mid sign_G(u,v) = +1\}|$. Similarily, we define $deg_-^+(v)$ and $deg_-^-(v)$.*

Let $G = (V_1 \cup V_2, E)$ be a bipartite graph such that V_1 and V_2 are independant sets satisfying $|V_1| = |V_2| = n \in \mathbb{N}$ (otherwise the answer to problem $\sharp P$M is trivially 0).

Let $H = (E, A)$ be the digraph such that $(e_1, e_2) \in A$ and $(e_2, e_1) \in A$ if and only if the edges e_1 and e_2 are adjacent in G.

Remark C.2 $|A| = \sum_{v \in V, deg(v) > 1} 2\binom{deg(v)}{2}$.

Let us show perfect matchings P of G corresond bijectively to possible signatures of H and in a way such that the set $W = P \subseteq E$ satisfies $|W| = |V_1| = |V_2| = n$ as well as $a = (e_1, e_2) \in A \cap (W \times V) \Leftrightarrow sign_{G^s}(a) = -$.

Let H^s be the signed digraph associated to H such that :

$$\forall e_1, e_2 \in E,\ sign_{H^s}(e_1, e_2) = \begin{cases} -1 & \text{if } e_1 \in P \text{ and } e_2 \in E \setminus P \\ +1 & \text{if } e_1 \in E \setminus P \text{ and } e_2 \in P \\ +1 & \text{if } e_1, e_2 \in P \end{cases}$$

Note that if e_1 and e_2 both belong to P then $(e_1, e_2) \notin A$ since by definition of P, they are non adjacent. Therfore, the only arcs $a \in A$ such that $sign_{H^s}(a) = -1$ are the outgoing arcs of a vertice representing an edge of P *i.e.,* the arcs between a vertice of H representing an edge $e_1 \in P$ and an other vertice representing an edge $e_2 \in E \setminus P$.

Also, by construction of H^s, a vertice which is the starting point of a negative arc has an outdegree equal to 0 in $\mathcal{RO}(H^s)$. Thus, in this graph, a $>$-arc cannot belong to a circuit. H^s is therefore a possible signed digraph.

Injectivity :
Let P_1 and P_2 be two perfect matchings of a same bipartite graph $G = (V_1 \cup V_2, E)$. Let $e \in P_1 \setminus P_2$. e must be adjacent to at least one other edge of E because if not, in order to cover its endings it must necessarily belong to all prefect matchings of G. So there

[4] *i.e.,* a set of edges covering all vertices and not containing any pair of adjacent edges.

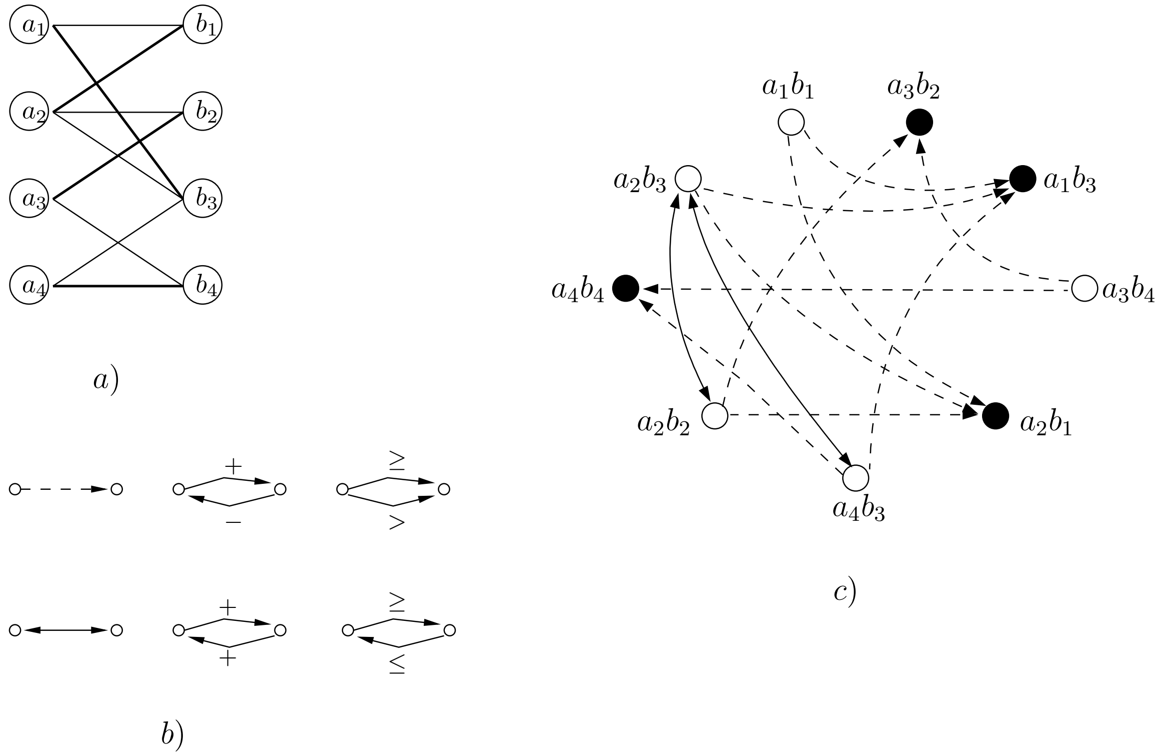


Figure 10: a. A bipartite graph $G = (V, E)$ such that $V = \{a_1, \ldots, a_4\} \cup \{b_1, \ldots, b_4\}$. The edges in bold form a perfect matching of G. b. On the left, the arcs used in figure c, in the center their meaning in terms of signed digraphs and on the right their meaning in terms of reoriented digraphs. c. The digraph resulting from the reduction of the instance of problem PM pictured in figure a.

exists an edge $e' \in E \setminus P$ which is adjacent to e. If H_1 is the signed digraph corresponding to P_1 and H_2 the one that corresponds to P_2, we find that $sign_{H_1}(e_1, e_2) = -1$ and $sign_{H_2}(e_1, e_2) = +1$. Thus, $H_1 \neq H_2$.

Surjectivity :
In addition, there is no possible signature of H that does not correspond to a perfect matching of G :

Let H^s be a possible signed digraph associated to H such that the set $W \in E$ satisfies the conditions of $\sharp G$S. If two edges e_1 and e_2 of W are adjacent, then $sign_{H^s}(e_1, e_2) = sign_{H^s}(e_2, e_1) = -1$ and necessarily, there is a prohibited cycle of length two between e_1 and e_2 in the reoriented graph of H^s. This contradicts the fact that H^s is possible. Moreover, every vertice of V is covered by an edge of W since this set contains n non adjacent edges and G is bipartite.

D Some observations on the equivalence classes $[\cdot]^D$

D.1 Ordinary boolean networks

Let $R_1 = (G, F, s_1)$ and $R_2 = (G, F, s_2)$ be two boolean networks with the same underlying digraph $G = (V, A)$ and whose local activation functions are symmetrical. In the sequel, we suppose that $G^{s_1} \neq G^{s_2}$.

Notations D.1

$$D = \{i \in V \ \mid \exists j \in N(i),\ sign_{s_1}(j,i) \neq sign_{s_2}(j,i)\},$$
$$t^* = min\{s_1(i) \mid i \in D\},$$
$$\forall i \in V, \quad N^+(i) = N^+_{s_1}(i) \cap N^+_{s_2}(i) \quad et \quad N^-(i) = N^-_{s_1}(i) \cap N^-_{s_2}(i),$$

By remark 3.2, $\forall i \notin D$ such that $s_1(i) \leq t^*$, $f_i^{s_1} = f_i^{s_2}$ since such vertices satisfy $\forall j \in N(i)$, $sign_{s_1}(j,i) = sign_{s_2}(j,i)$.

Also, $\forall i \in D$, $\forall x \in \{0,1\}^n$,

$$\begin{aligned} f_i^{s_1}(x) &= f_i(y_i(x)\ ;\ x_j \mid j \in N^+_{s_1}(i) \cap N^-_{s_2}(i)\ ;\ f_j^{s_1}(x) |\ j \in N^-_{s_1}(i) \cap N^+_{s_2}(i)) \\ f_i^{s_2}(x) &= f_i(y_i(x)\ ;\ f_j^{s_2}(x) |\ j \in N^+_{s_1}(i) \cap N^-_{s_2}(i)\ ;\ x_j \mid j \in N^-_{s_1}(i) \cap N^+_{s_2}(i)) \end{aligned}$$

where $y_i(x) = N^+(i) \ \cup \ N^-(i)$. Therfore, in the case where $F^{s_1} = F^{s_2}$, the two expressions above must be equal. And in the specific case where all local activation functions are equal to the function f, these two expressions must even be equal $\forall i \in V$.

If R_1 and R_2 share a dynamical cycle $C = (x^0, \dots, x^{l-1}, x^l = x^0)$, then any $i \in D$ such that

- $N^+_{s_1} \cap N^-_{s_2} = \emptyset$ and $N^-_{s_1} \cap N^+_{s_2} \neq \emptyset$ *i.e.*, $N^+(i) = N^+_{s_1} \subset N^-_{s_2}$ or
- $N^-_{s_1} \cap N^+_{s_2} = \emptyset$ and $N^+_{s_1} \cap N^-_{s_2} \neq \emptyset$ *i.e.*, $N^-(i) = N^-_{s_1} \subset N^+_{s_2}$

satisfies $x_i^{t+1} = x_i^t$, $\forall t \in [\![0, l[\![$.

D.2 R_p and other boolean networks

Here, we study the equivalence class $[R_p]^D$, where $R_p = (G, F, s_p)$. In particular we are interested in finding what other boolean networks may belongs to this class.

Notations D.2 *For a given boolean network, $R = (G, F, s)$, and for any set E of points $x \in \{0,1\}^n$, we write $C^{st}(E) = \{i \in V(G) \mid \forall x \in E,\ f_i^s(x) = x_i\}$. We also define $D = \{i \in V \mid \exists j \in N(i),\ sign_s(j,i) = -\}$.*

The following theorem taken from [6] partially answers the question :

Theorem D.3 *Let $G=(V,A)$ be a digraph without loops*[5] *and F a local activation function. Then, if s is a sequential update schedule, $R=(G,F,s)$ and R_p cannot share the same dynamical cycles.*

Preuve Let $C=(x^0,\dots,x^{l-1},x^l=x^0)$ be a dynamical cycle such that $\forall t\in [\![0,l[\![$, $\forall i\in V$, $x^{t+1}=f_i^s(x^t)=f_i^{s_p}(x^t)$. Let i^* be a vertice such that $s(i^*)=max\{s(i)\mid \exists t\in [\![0,l[\![,\ x_i^{t+1}\neq x_i^t\}$. Then

$$\begin{array}{rcl} x_{i^*}^{t+1} & = & f_{i^*}^{s_p}(x^t)\\ & = & f_{i^*}(x_j^t \mid j\in N(i^*)\cap C^{st}(C)\ ;\ x_j^t \mid j\in N(i^*)\setminus C^{st}(C))\\ & = & f_{i^*}^{s}(x^t)\\ & = & f_{i^*}(x_j^t \mid j\in N(i^*)\cap C^{st}\ ;\ x_j^t \mid j\in N_s^+(i^*)\setminus C^{st}(C)=\emptyset\ ;\ x_j^{t+1}\mid j\in N_s^-(i^*)\setminus C^{st}(C))\\ & = & f_{i^*}(x_j^t \mid j\in N(i^*)\cap C^{st}(C)\ ;\ x_j^{t+1}\mid j\in N_s^-(i^*)\setminus C^{st}(C))\\ & = & f_{i^*}^{s_p}(x^{t+1})\end{array}$$

We obtain a contradiction with the fact that $x_{i^*}^t\notin C^{st}(C)$. □

Remark D.4 *If s is block sequential such that $(N(i^*)\setminus C^{st}(C))\cap B_{s(i^*)}^s=\emptyset$ or if $B_{s(i^*)}^s\setminus C^{st}(C)=\{i^*\}$, the proof above and theorem D.3 still hold.*

Let us now look at the case where s is block sequential.

1. For all $i\in V\setminus D$, $N_s^-(i)=\emptyset$ and $f_i^s=f_i^{s_p}$ (*cf* remark 3.2).

2. Let $E\subseteq\{0,1\}^n$ and $i\in D$ such that $N_s^-(i)\subseteq C^{st}(E)$.

   Then, $\forall x\in E$,

   $$\begin{array}{rcl} f_i^s(x) & = & f_i(x_j\mid j\in N_s^+(i)\ ;\ f_j^s(x)=x_j\mid j\in N_s^-(i))\\ & = & f_i(x_j\mid j\in N(i))\\ & = & f_i^{s_p}(x).\end{array}$$

   In particular, if in G^s all vertices i that are the starting point of a negative arc have null indegree or are such that $N(i)=\{i\}$, then $R_s \overset{D}{\sim} R_p$. However, to know whether there exists other interesting cases of networks such that $\forall i\in D,\ N_s^-(i)\subseteq C^{st}(E)$, we would need to study the situations in which a vertice can have a constant state.

3. Let $E\subseteq\{0,1\}^n$ and $i\in D$ such that $N_s^-(i)\setminus C^{st}(E)\neq\emptyset$. Then

   $f_i^s(x)=f_i(x_j\mid j\in C^{st}(E)\cap N(i)\ ;\ x_j\mid j\in N_s^+(i)\setminus C^{st}(E)\ ;\ f_j(x)\mid j\in N_s^-(i)\setminus C^{st}(E))$.

   If furthermore $N_s^+(i)\subseteq C^{st}(E)$ and $\forall x\in E$, $F^s(x)=F^{s_p}(x)$ then $i\in C^{st}(E)$. Indeed, $\forall x\in E$ the following holds

   $$\begin{array}{rcl} f_i^s(x) & = & f_i(x_j\mid j\in C^{st}(E)\cap N(i)\ ;\ f_j^s(x)\mid j\in N_s^-(i)\setminus C^{st}(E))\\ = f_i^{s_p}(x) & = & f_i(x_j\mid j\in C^{st}(E)\cap N(i)\ ;\ x_j^t\mid j\in N_s^-(i)\setminus C^{st}(E)).\end{array}$$

[5] In reality, without *n*on monotonous loops is enough.

This implies that $\forall x \in E$, $f_i^s(x) = f_i^{s_p}(F^s(x)) = f_i^{s_p}(x) = x_i$. Thus, if for all vertice i of D, $N_s^-(i)\setminus C^{st}(E) \neq \emptyset$ and $N_s^+(i) \subseteq C^{st}(E)$, then $R_s \overset{D}{\sim} R_p$ and the only possible limit behaviors that these two networks can have are fixed points and cycles such that the only vertices i satisfying $N(i) = N^+(i)$ have a non constant state.

4. Let $E \subseteq \{0,1\}^n$. Suppose that for all $i \in D$ such that $N_s^-(i) \setminus C^{st}(E) \neq \emptyset$, f_i is associative, *i.e.*, $\forall E_1, E_2 \subseteq V$

$$f_i(x_j \mid j \in E_1 \; ; \; x_j \mid j \in E_2) = f_i(f_i(x_j \mid j \in E_1) \; ; \; f_i(x_j \mid j \in E_2)),$$

Suppose also that

$$f_i(x_j \mid j \in N_s^-(i) \setminus C^{st}(E)) = f_i(f_j(x) \mid j \in N_s^-(i) \setminus C^{st}(E)).$$

In these conditions and if $E = \{0,1\}^n$, $R_s \overset{D}{\sim} R_p$.

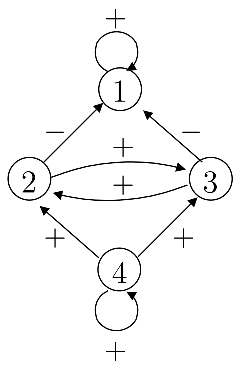


Figure 11:

Figure 4 represents a boolean network $R = (G, F, s)$ associated to a graph of order 4. The set D in this example is reduced to vertice 1. Also, the vertices 2 and 3 which belong to $N_s^-(1)$ have a non constant state. Suppose all local activation functions are equal to the same symmetrical function whose infix notation is $\star$. Then

$$\begin{aligned}
f_1^{s_p} &= x_1 \star x_2 \star x_3 \\
f_1^s &= x_1 \star f_2^s(x) \star f_3^s(x) \\
f_2^{s_p} &= f_2^s = x_3 \star x_4 \\
f_3^{s_p} &= f_3^s = x_2 \star x_4 \\
f_4^{s_p} &= f_4^s = x_4
\end{aligned}$$

If $\star = \vee$ (OR function), then since $4 \in N(2) \setminus N(1)$, x_4 appears in the development of $f_1^s(x)$ but not in that of $f_1^{s_p}(x)$:

$$f_1^{s_p} = x_1 \vee x_2 \vee x_3 \neq f_1^s = x_1 \star (x_3 \vee x_4) \vee (x_2 \vee x_4) = x_1 \vee x_2 \vee x_3 \vee x_4$$

and thus $R \notin [R_p]^D$. If $\star = \oplus$ (XOR function), then on the contrary, $R \in [R_p]^D$ because x_4 appears a pair number of times in the development of the expression $f_1^s(x)$:

$$f_1^{s_p} = x_1 \oplus x_2 \oplus x_3 = f_1^s = x_1 \star (x_3 \oplus x_4) \oplus (x_2 \oplus x_4).$$